Advanced features in Score-P and Scalasca

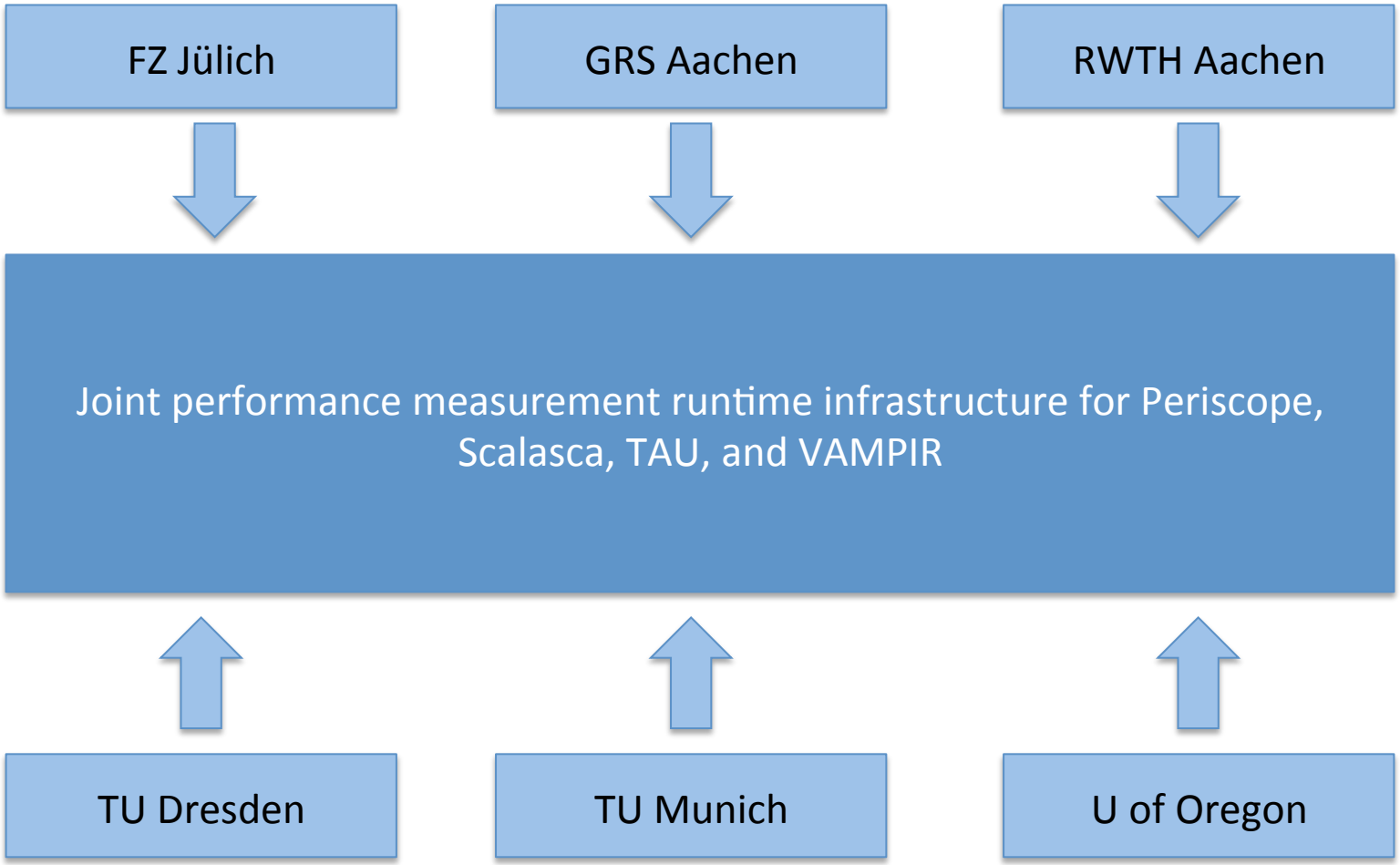
David Böhme, LLNL

Petascale Tools Workshop 2014

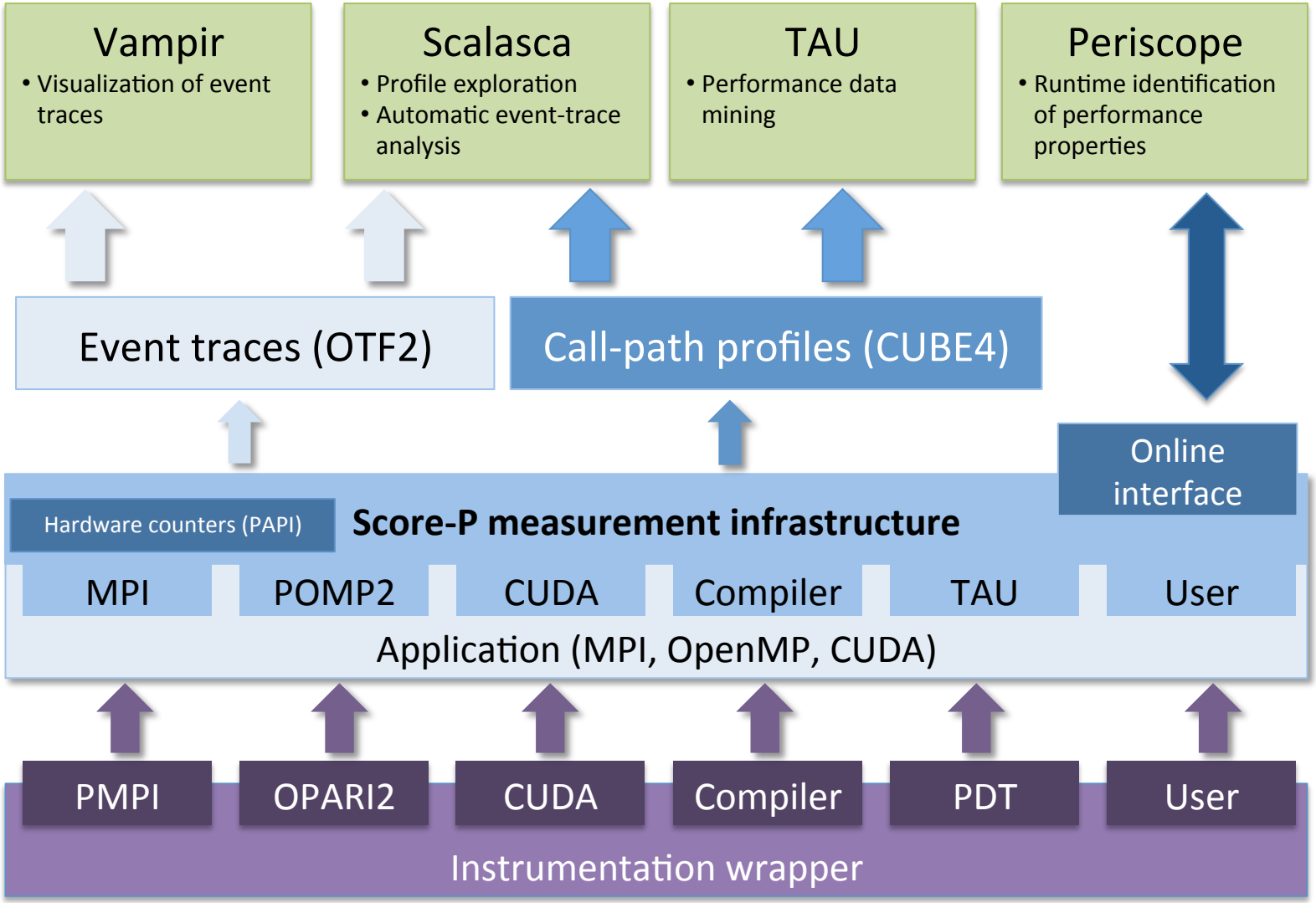
Outline

- Score-P and Scalasca overview
- Feature highlights
  - Scalable time-series profiles
  - Critical-path analysis
  - Delay analysis

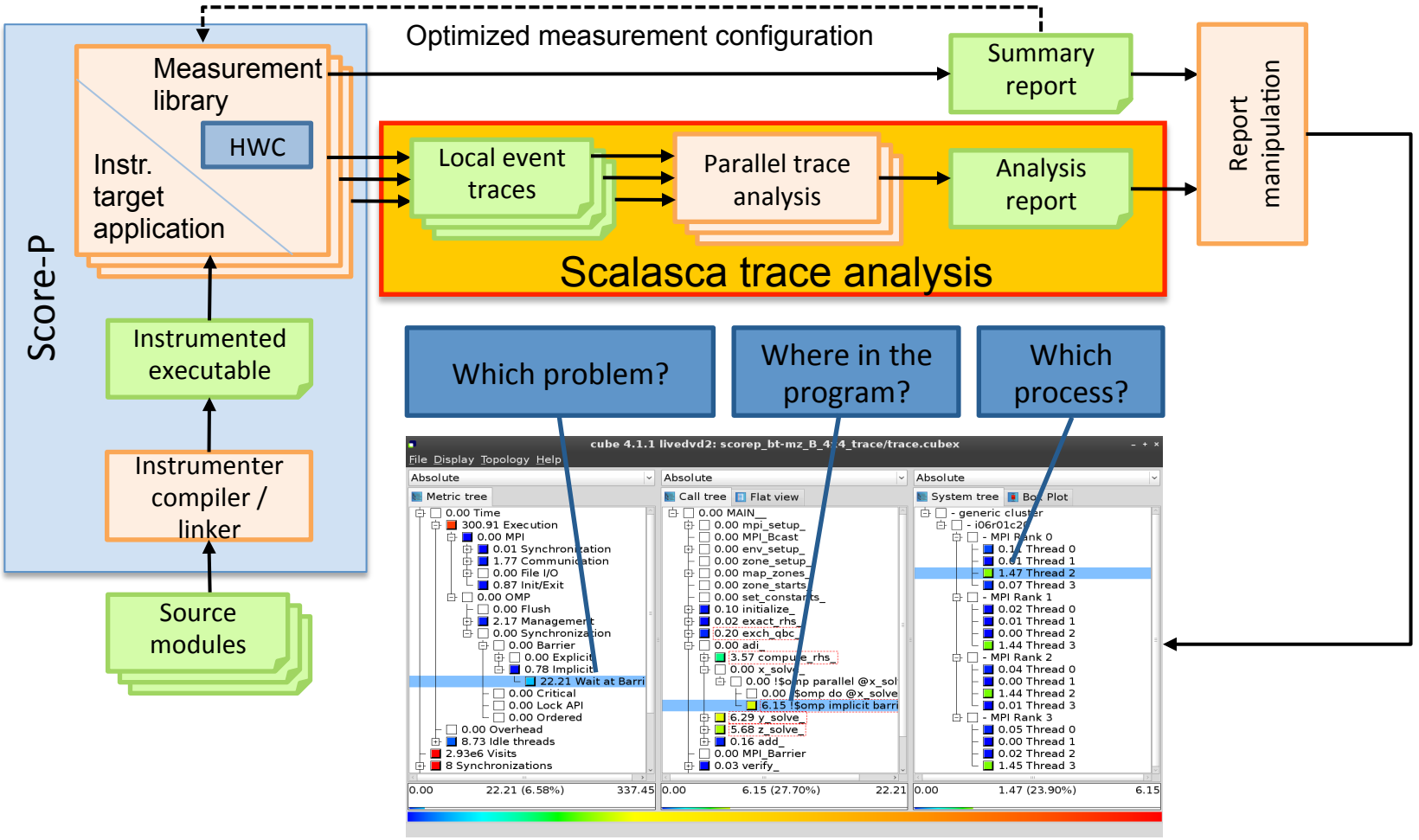
Score-P


FZ Jülich
GRS Aachen
RWTH Aachen
Joint performance measurement runtime infrastructure for Periscope, Scalasca, TAU, and VAMPIR
TU Dresden
TU Munich
U of Oregon

Score-P tool suite architecture

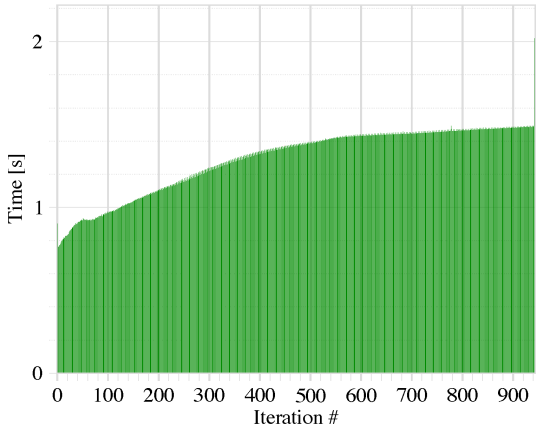


Scalasca analysis workflow

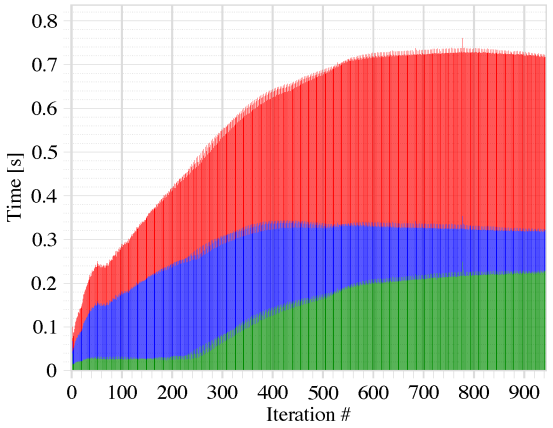


New features capture performance dynamics

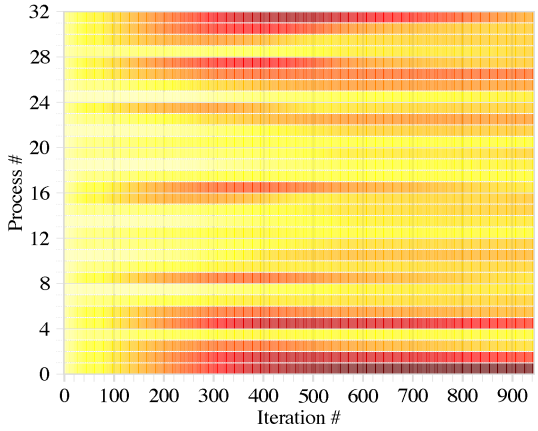
- Efficient time-series profiles in Score-P
- Critical-path and root-cause analysis in Scalasca



Execution time



Maximum, median, and minimum of point-to-point communication time



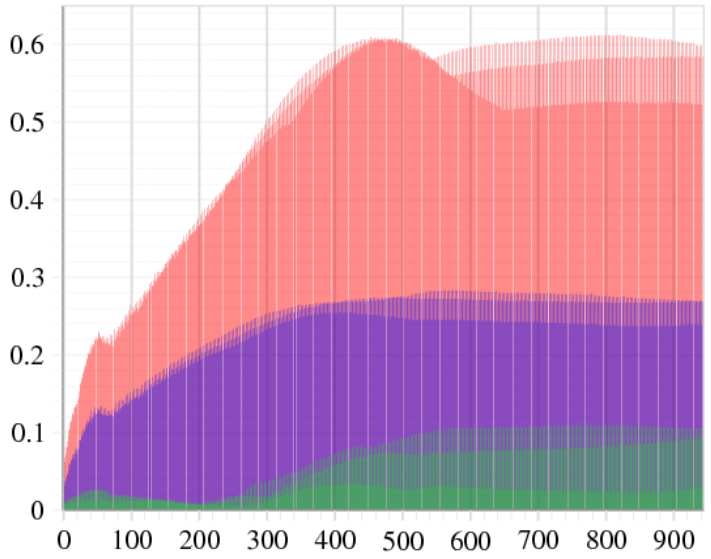
Complete distribution of point-to-point communication time

Time-dependent performance in 129.tera_tf (3D Eulerian hydrodynamics)

Time-series profiling in Score-P

- Mark start and end of main loop
- Record separate profile for each iteration

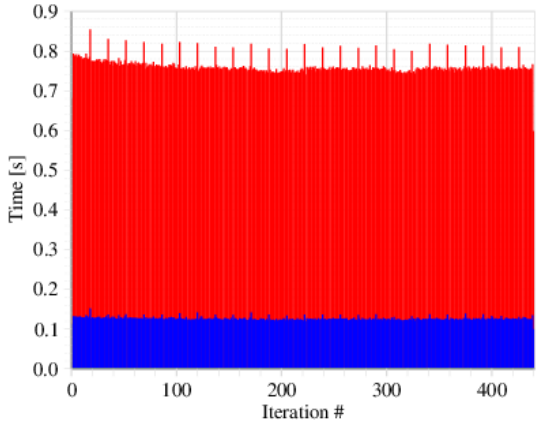
```
#include "scorep/SCOREP_User.h"

int main() {
  SCOREP_USER_REGION_DEFINE ( iter );

  initialize();
  read_input();

  for (int t = 0; t < 5; ++t) {
    SCOREP_USER_REGION_BEGIN( iter, "iter",
      SCOREP_USER_REGION_TYPE_DYNAMIC );

    do_work();
    do_additional_work();
    finish_iteration();

    SCOREP_USER_REGION_END( iter );
  }

  write_output();
  return 0;
}
```



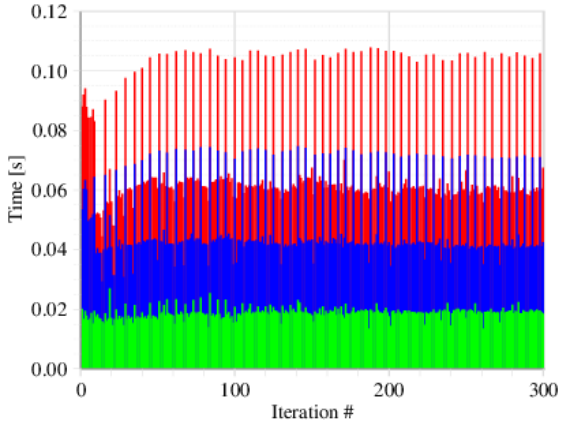
MPI point-to-point communication time

Examples from the SPEC MPI 2007 benchmark suite

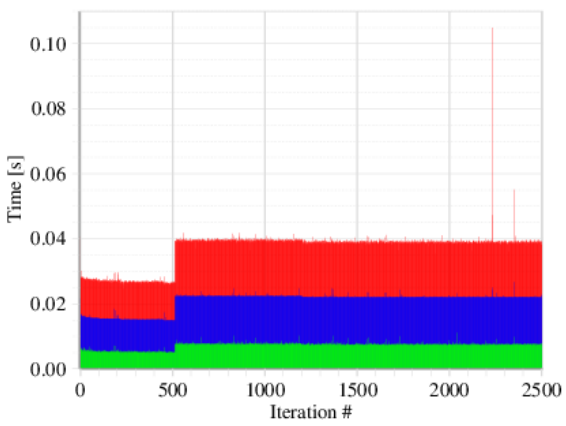
121.pop2



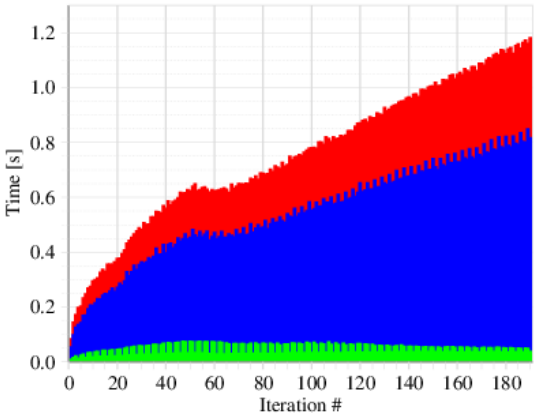
126.lammps



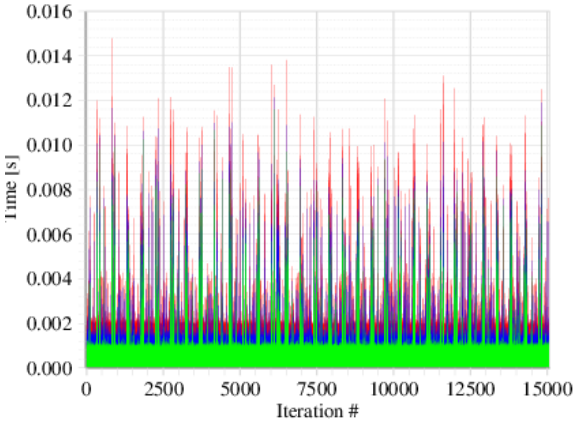
128.GAPgeofem



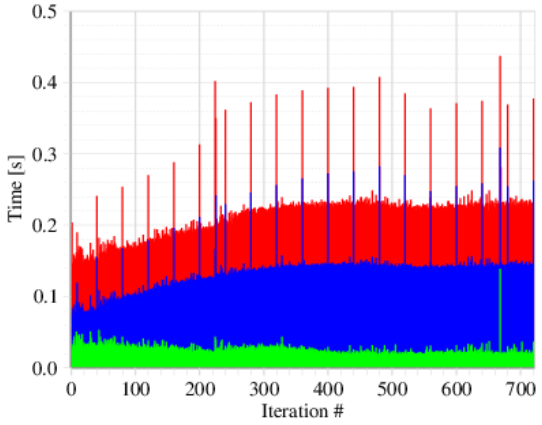
129.tera_tf



143.dleslie

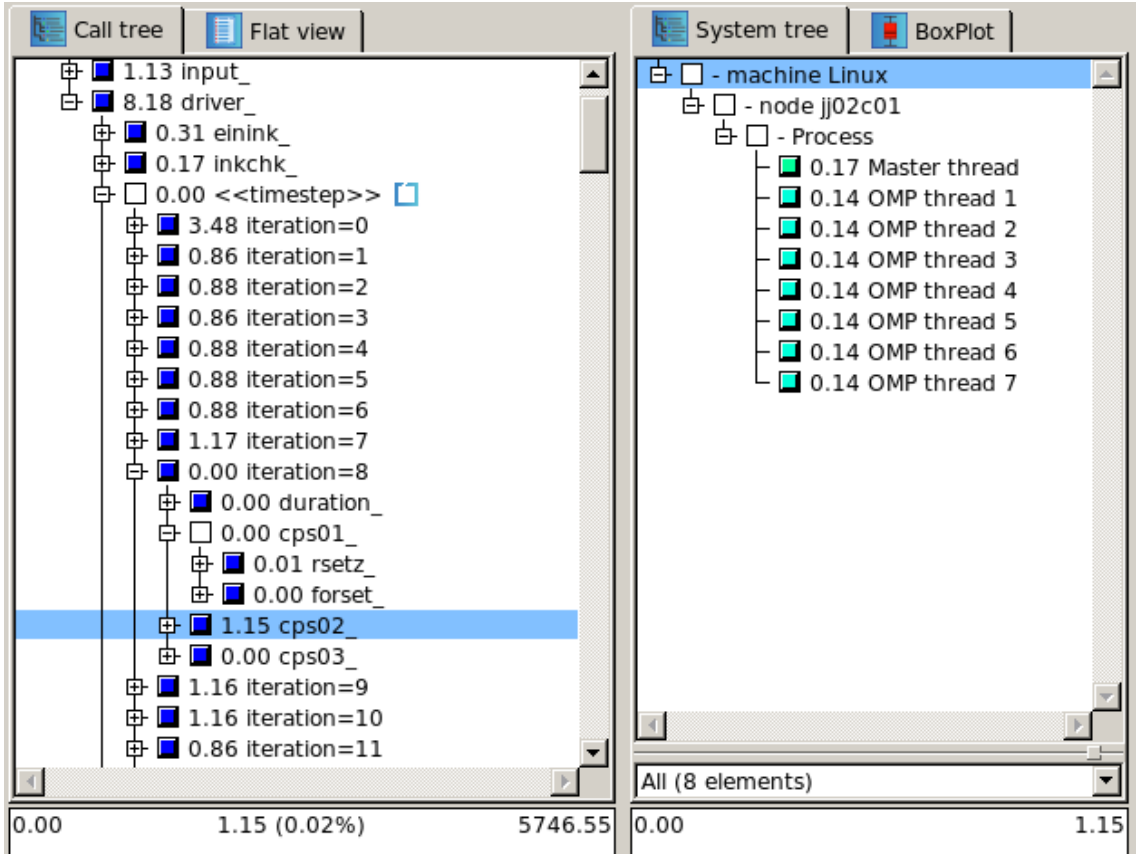


147.l2wrf2



MPI point-to-point communication time

A separate call-tree is created for every iteration



Call tree

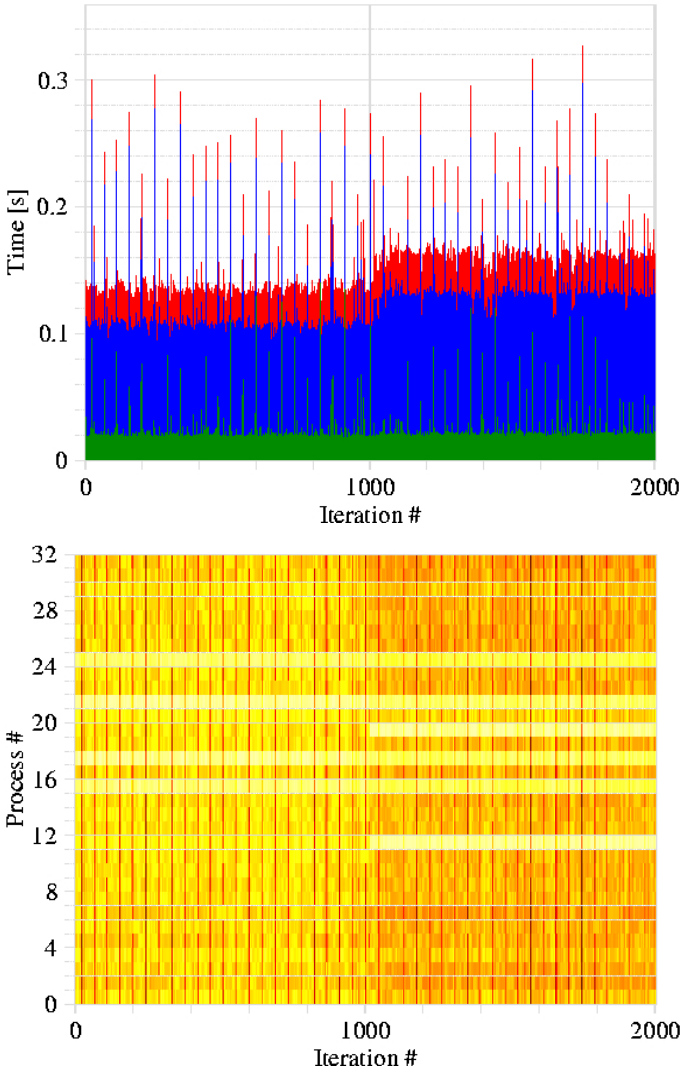
Process topology

Storage challenge: amount of data proportional to the number of iterations

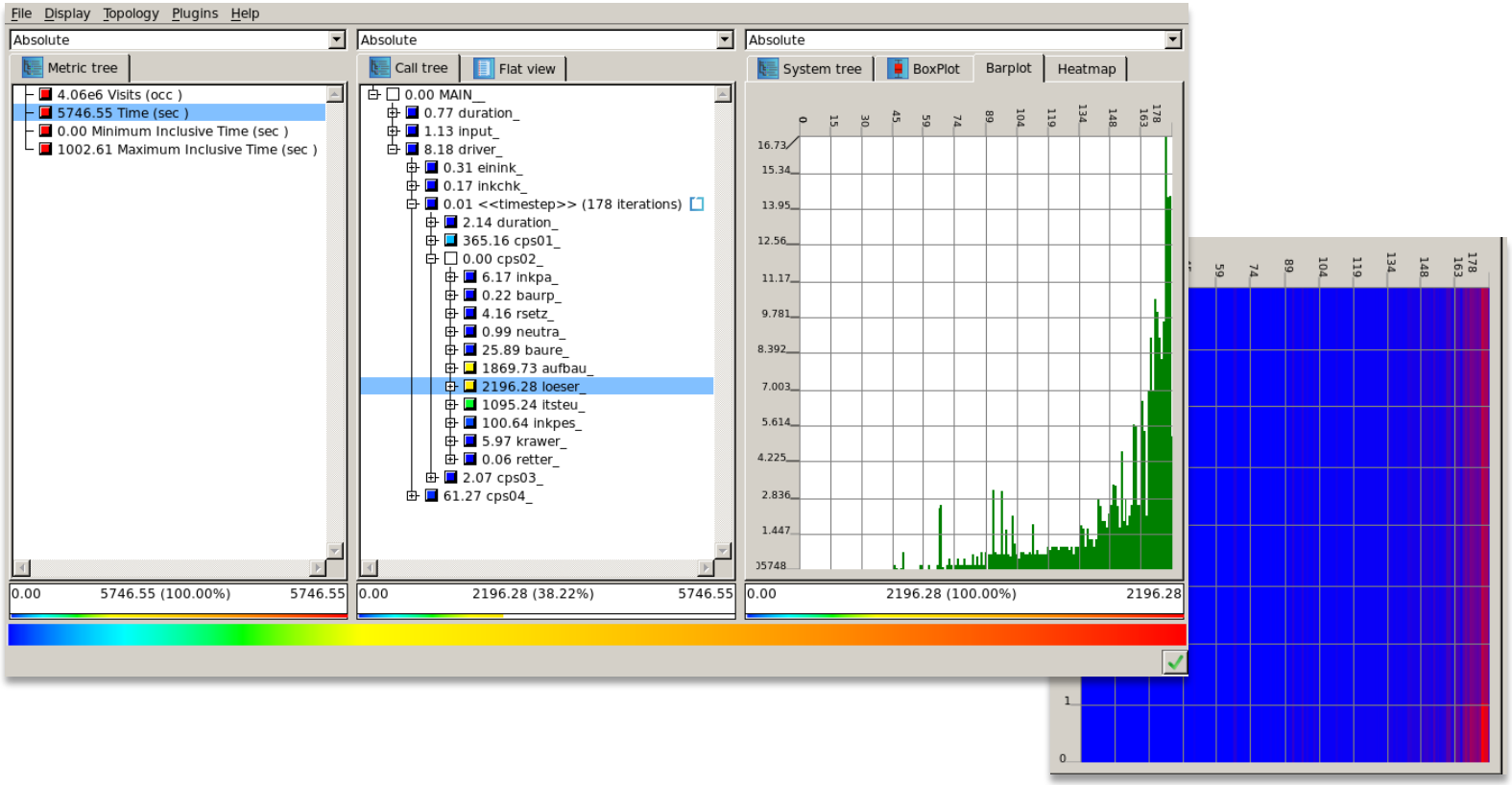
Incremental on-line compression

- Exploits that many iterations are very similar
  - Summarizes similar iterations in a single data element via clustering
- On-line to save memory at run-time
- Process-local to
  - Avoid communication
  - Adjust to local temporal patterns
- The number of clusters can never exceed a predefined maximum
  - Merging of the two closest ones
- Available since Score-P 1.1 (10/2012)

MPI point-to-point time in 107.leslie3d



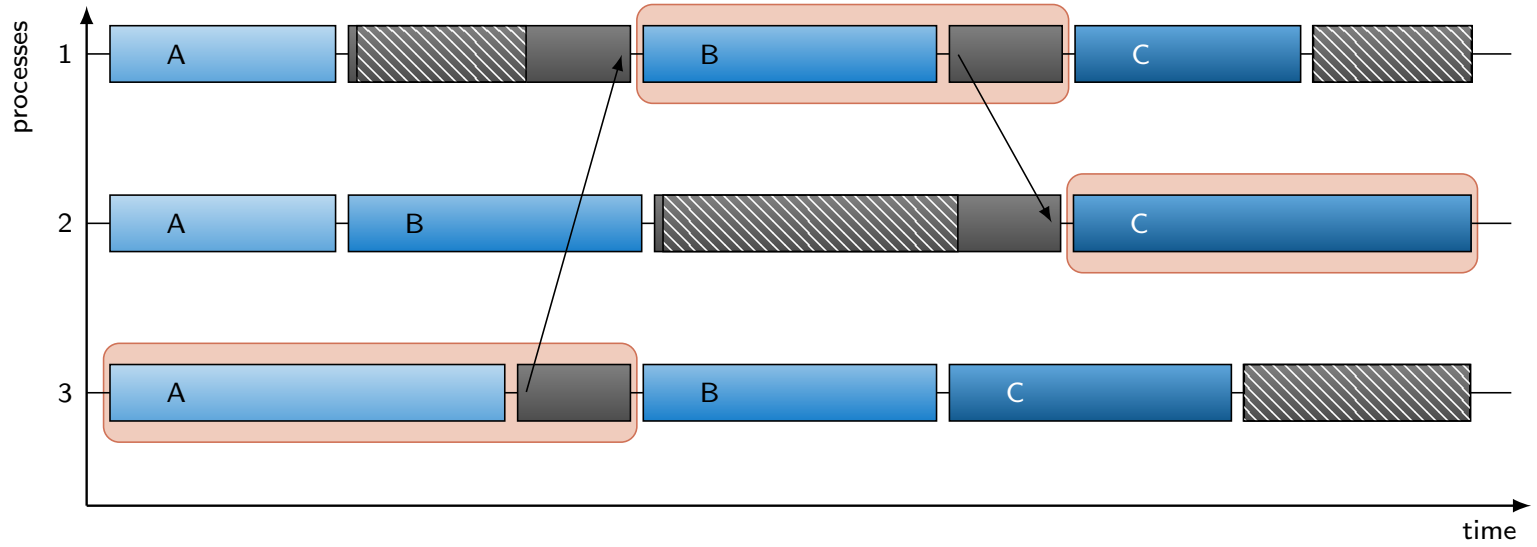
Barplot and Heatmap display in Profile Browser



INDEED time-series profile / 8 Threads on Juropa

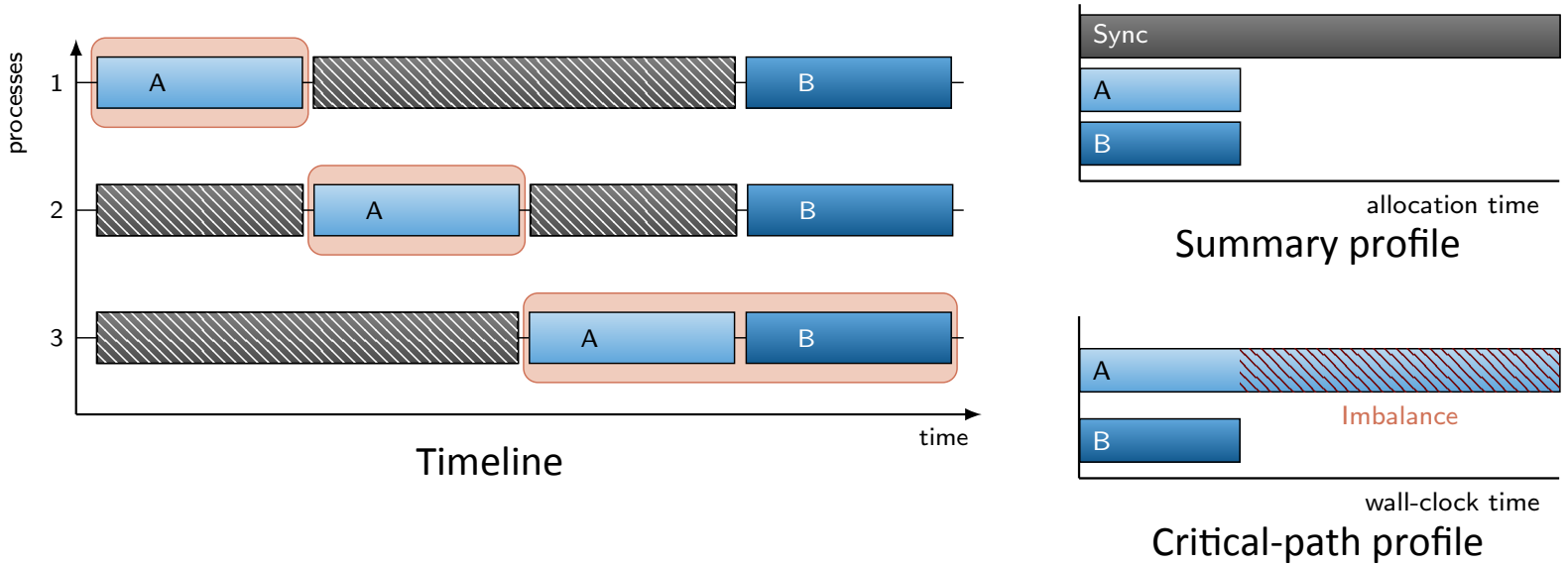
Critical-path analysis in Scalasca

- Use automatic trace analysis to extract the critical path
- Performance indicators highlight parallel bottlenecks



Critical path in a parallel program (shown in red)

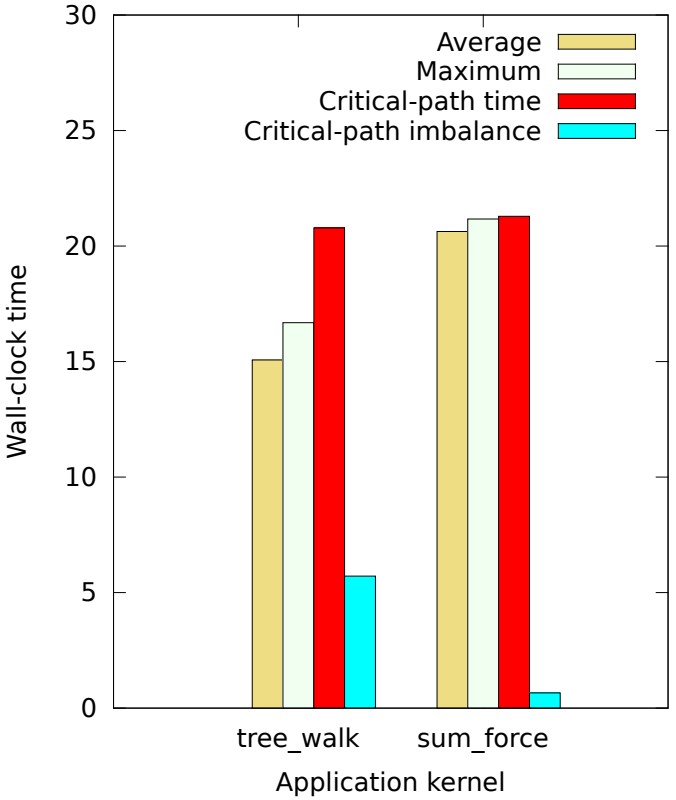
Identifying dynamic load imbalance



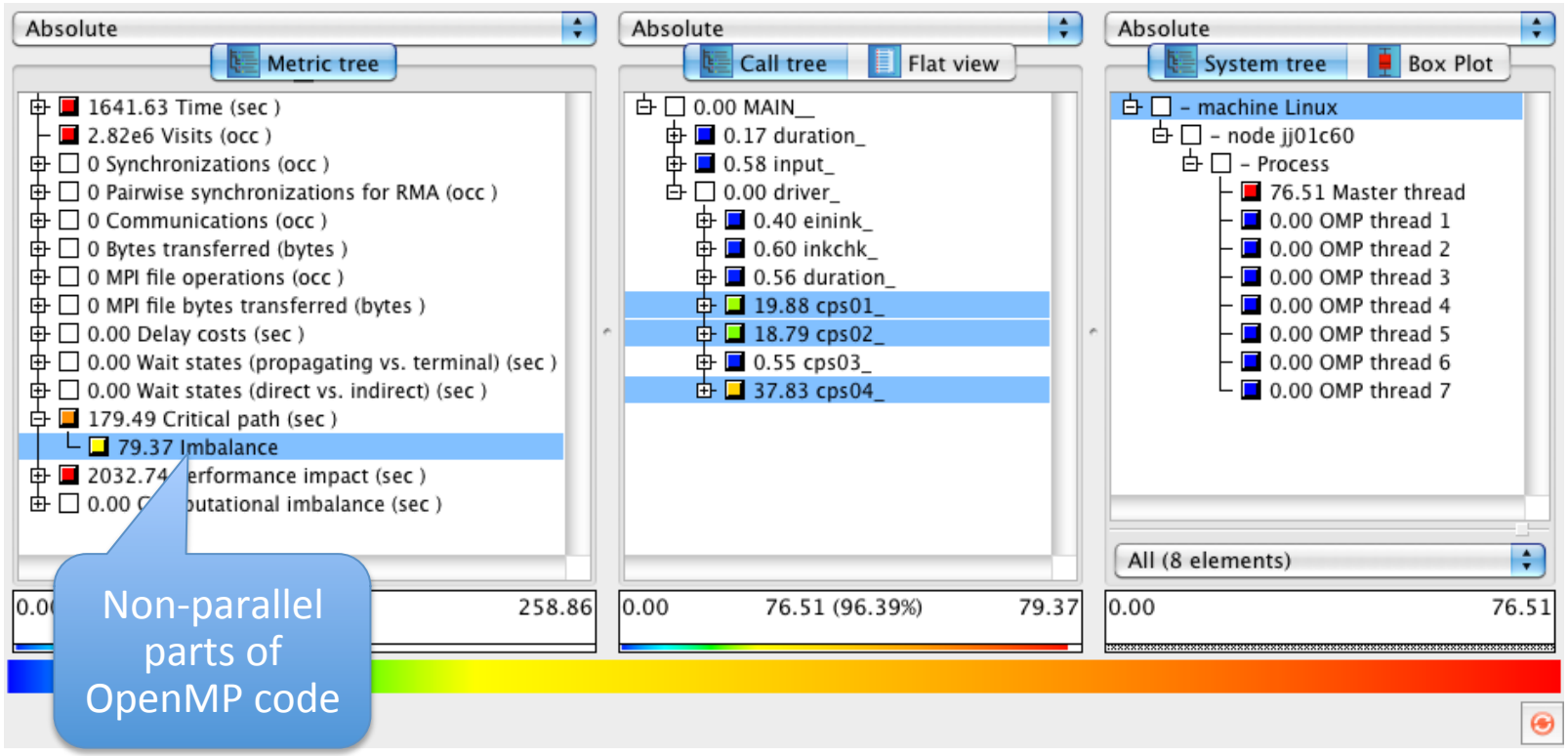
- Critical-path profile shows wall-clock time consumption
- Critical imbalance indicator finds inefficient parallelism
  - Imbalance = $T_{critical} - T_{average}$

Example: PEPC

- Analysis of plasma-physics code PEPC using 512 processes on Blue Gene/P
- Profile metrics underestimate performance impact of `tree_walk` kernel due to dynamic load imbalance



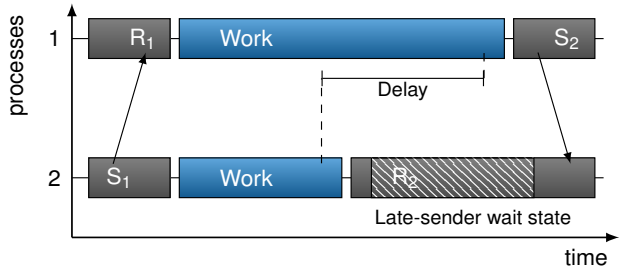
Critical path in Cube display



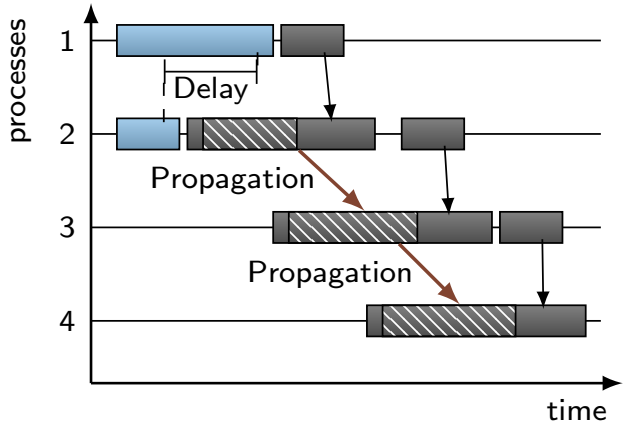
INDEED critical-path analysis / 8 OpenMP Threads on Juropa

Delay analysis finds root causes of wait states

- Identifies delays that cause wait states
- Assigns costs that represent total amount of waiting time caused
- Incorporates long-distance propagation effects

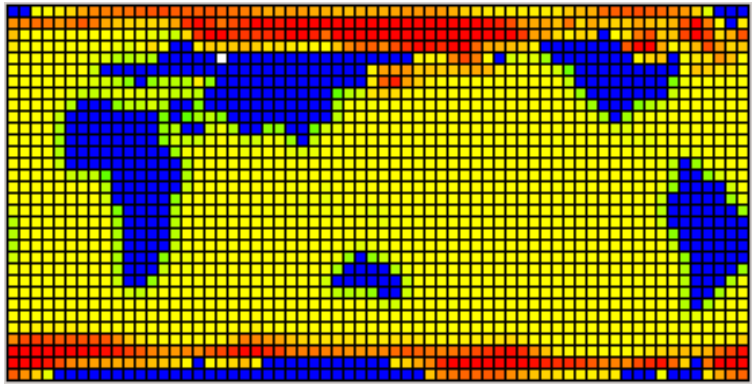


Delay on process 1 causes wait state

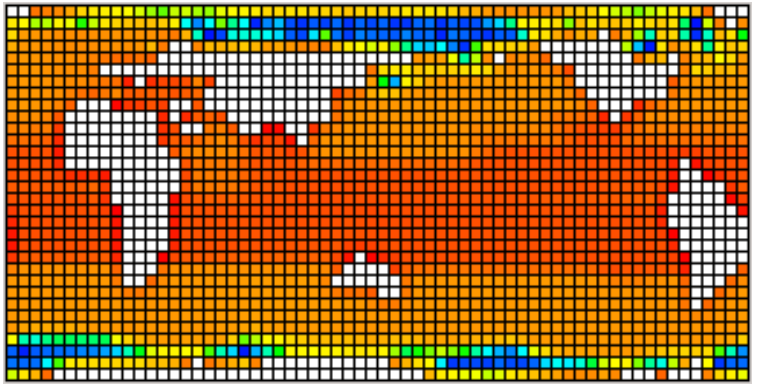


Delay costs account for propagation effects

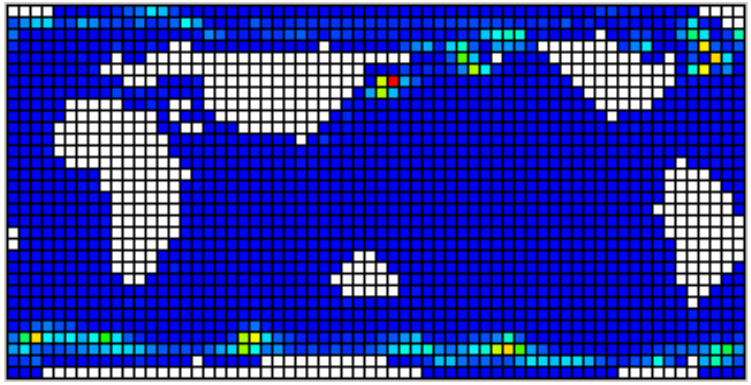
Example: CESM sea ice model



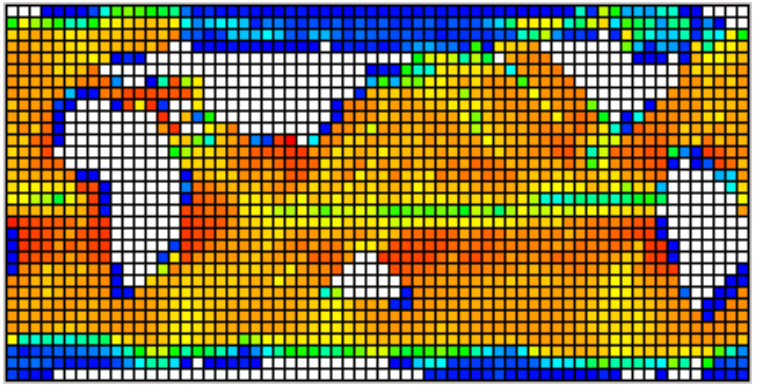
Distribution of computation time



Distribution of late-sender waiting time



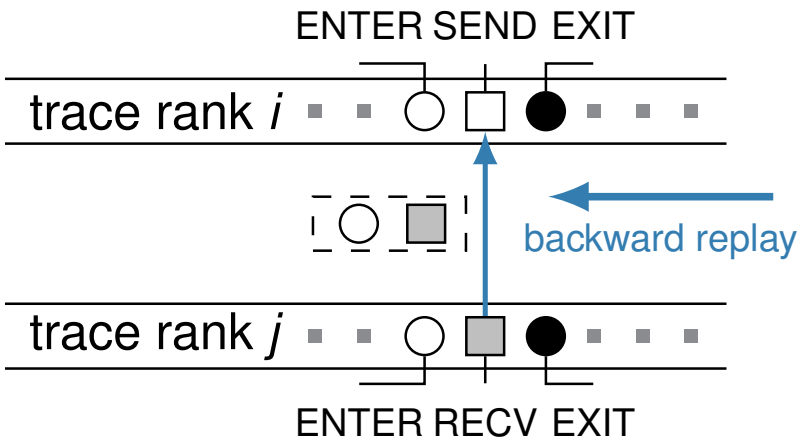
Distribution of delay costs



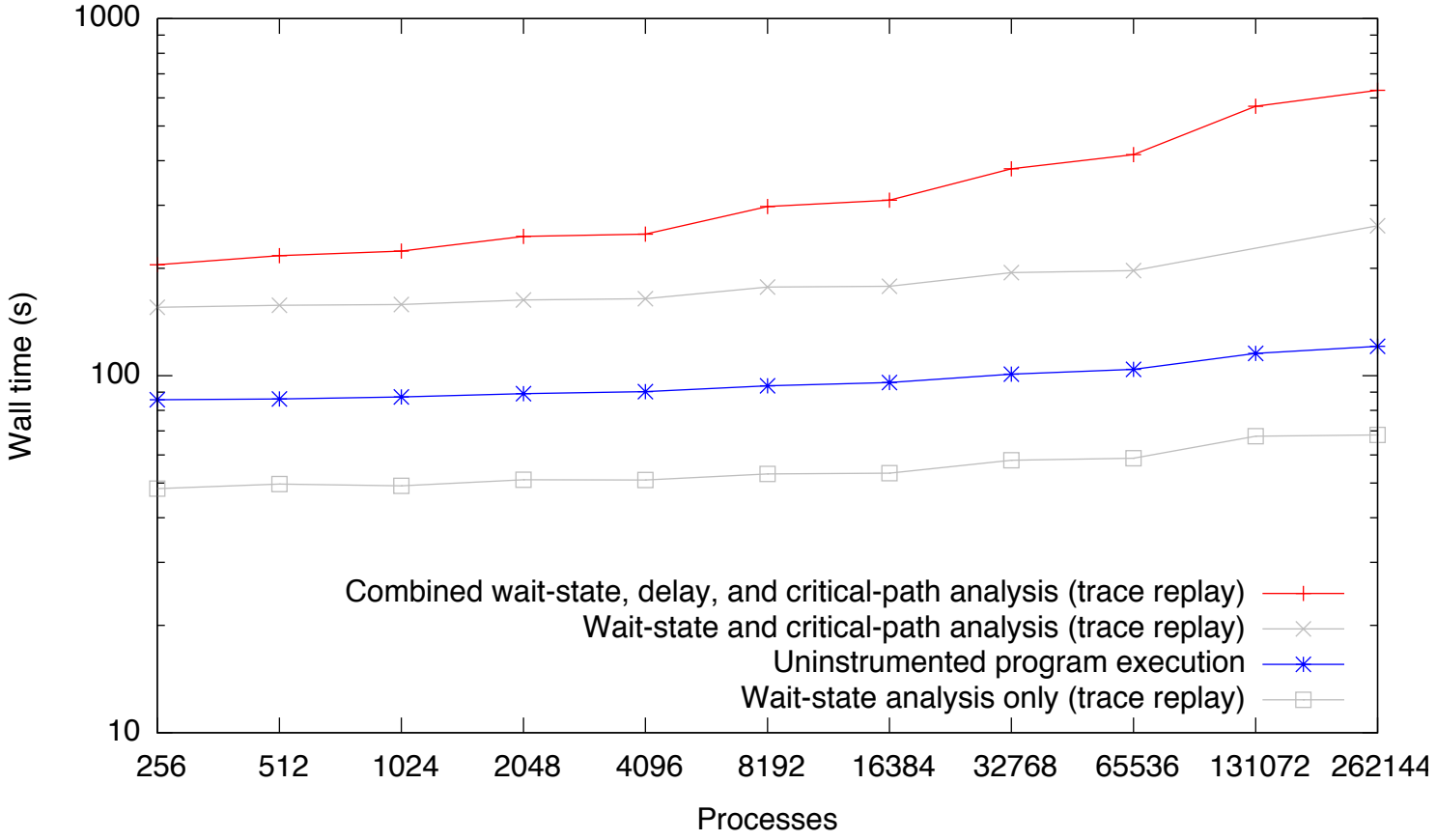
Distribution of propagating wait time

Trace analysis extension: backward replay

- Use multiple replay passes
- Backward replay lets data travel from effect to source



Scalability



Scalability of delay and critical-path analysis for the Sweep3D benchmark on Blue Gene/P

Outlook

- Score-P
  - Support for more programming models
  - Sampling
  - Power measurement
  - Scalable profile format
- Scalasca2
  - Task and lock-contention analysis